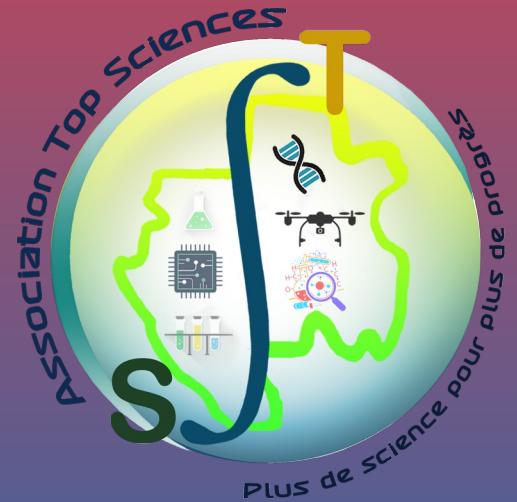


Régression logistique

Présenté par : Dr Nzamba Bignoumba

$$f(x) = \frac{1}{1 + e^{-x}}$$

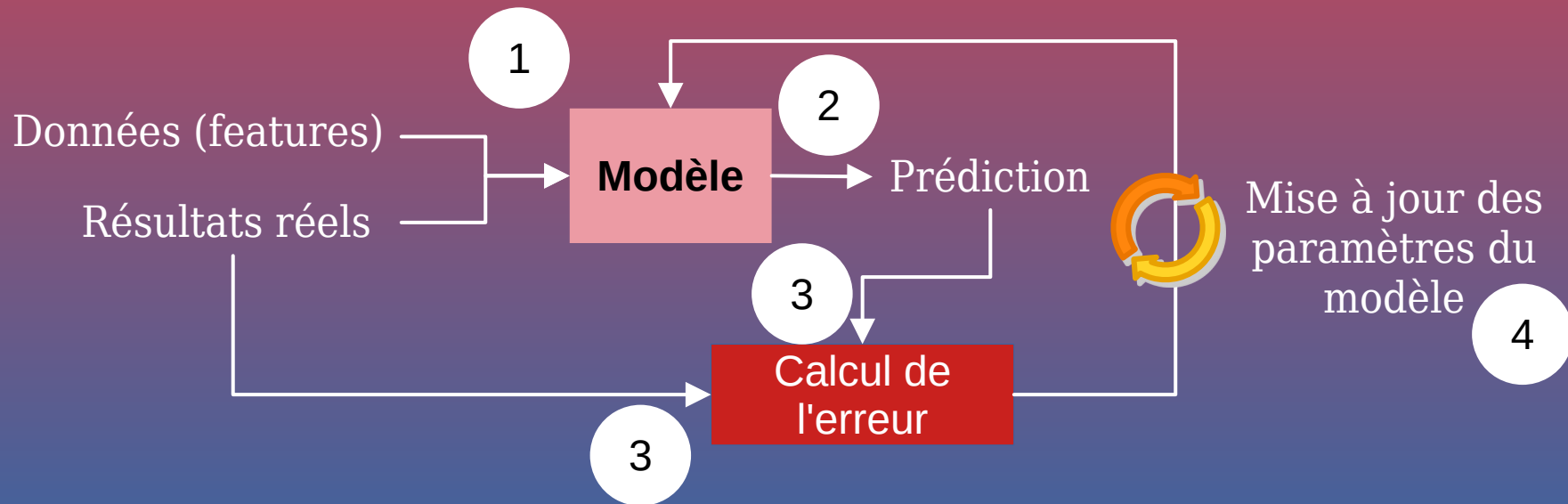


Outline

Introduction	→0h30
Théorie	→1h30
Etude de cas	→2h00
Déploiement des modèles	→1h00

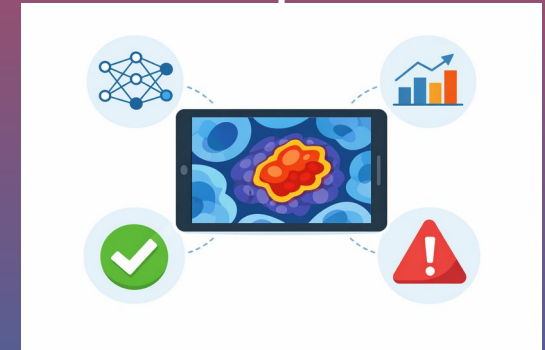
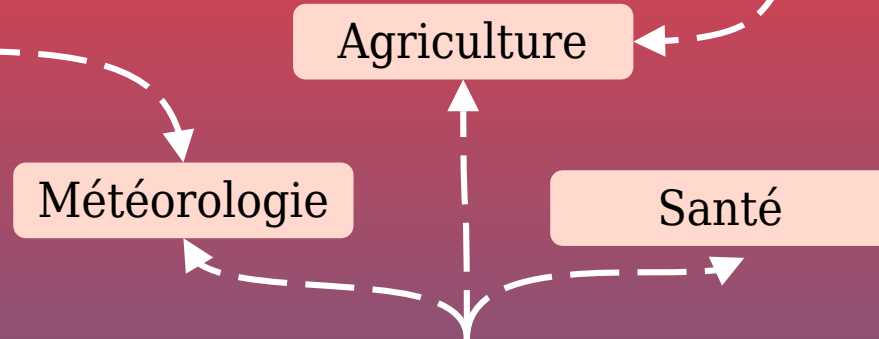
Introduction : définition

La **régression logistique** est un **algorithme supervisé de machine learning** qui permet de résoudre des **problèmes de classification binaire**. Un problème de classification binaire est un problème avec **deux résultats possibles** : 0 ou 1, vrai ou faux, accepté ou refusé, bon ou mauvais, etc.



Architecture d'un entraînement supervisé

Introduction : cas d'utilisation



Théorie

Supposons que nous voulions déterminer si un client **effectuerait ou non** un achat sur notre site web.

Pour cela, nous disposons de données telles que l'âge du client, le nombre d'heures passées sur le site et l'action finale réalisée (achat ou pas-achat).

Formulation des données

$$\mathbf{x}^T = [\text{age}, \text{tempspasse}] = [x_1, x_2]; y = 0/1$$

Où $0 \rightarrow \text{Pas achat}$; $1 \rightarrow \text{Achat}$

Objectif : Quelle est la probabilité (likelihood) qu'un client effectue un achat en fonction de son âge et du temps passé sur le site web ?

Objectif formel :

$$P(y|\mathbf{x}; \theta) = P(y|x_1, x_2; \theta) = ?$$

$\theta \rightarrow \text{Paramètres du modèle}$

age	temps_passe	probabilite	achat_seuil_0_5
32.0	6.41	0.89	1.0
22.0	4.34	0.45	0.0
29.0	6.38	0.85	1.0
38.0	4.46	0.81	1.0
23.0	9.8	0.96	1.0
33.0	5.83	0.86	1.0
24.0	6.1	0.74	1.0
24.0	8.83	0.94	1.0
34.0	8.55	0.97	1.0
31.0	9.14	0.97	1.0
18.0	8.6	0.88	1.0
35.0	1.32	0.33	0.0
21.0	9.71	0.95	1.0
34.0	5.52	0.85	1.0
27.0	4.45	0.59	1.0
29.0	7.71	0.93	1.0
31.0	3.21	0.51	1.0
33.0	2.34	0.43	0.0
29.0	8.72	0.96	1.0
23.0	7.84	0.88	1.0

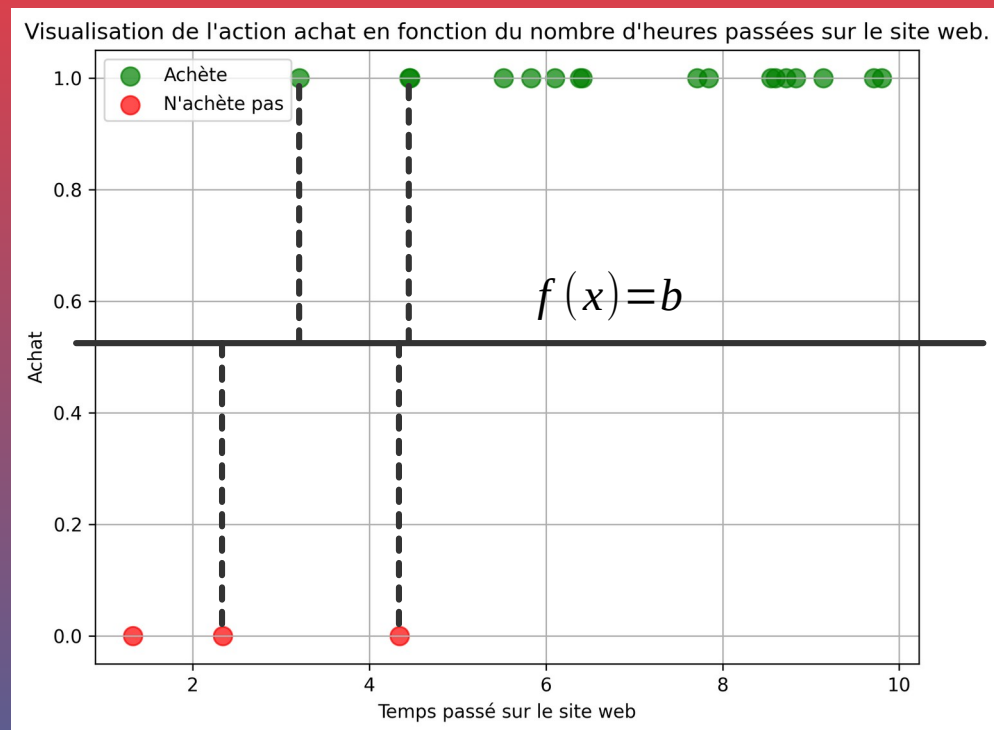
Visualisons les données en ne considérant que le temps passé sur le site web.

Cherchons une fonction séparatrice, c'est-à-dire notre **classificateur**.

Choix 1 : **fonction linéaire**



Si nous utilisons cette fonction, tous les points auront la même valeur. Par conséquent, ils appartiendront à la même classe. Ce qui n'est pas le cas.



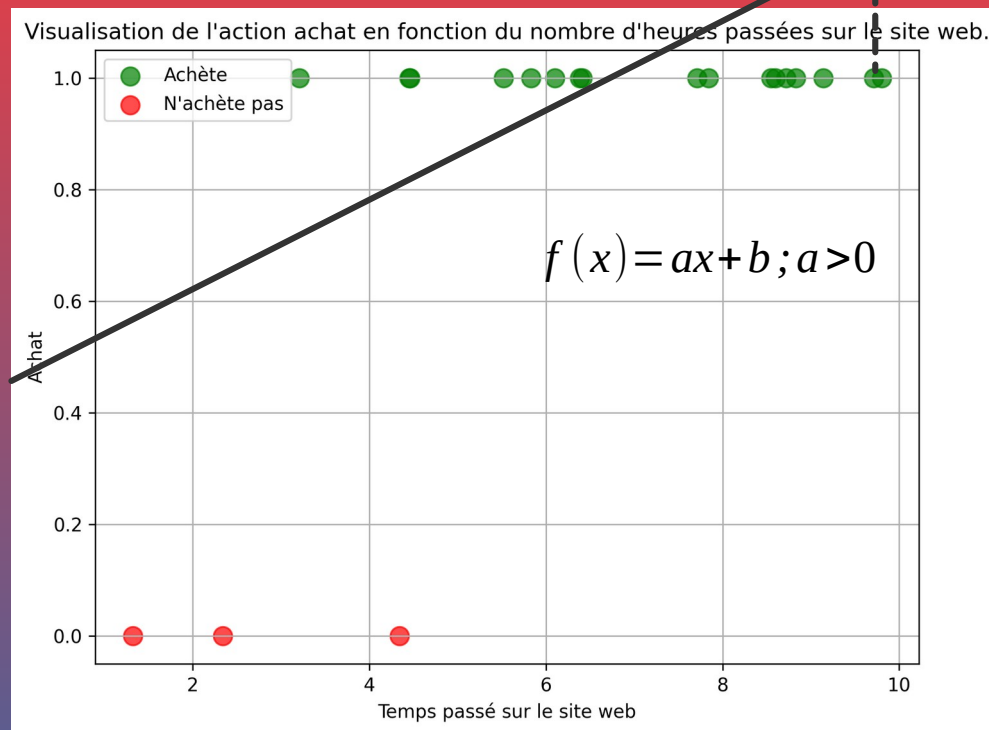
Visualisons les données en ne considérant que le temps passé sur le site web.

Cherchons une fonction séparatrice, c'est-à-dire notre **classificateur**.

Choix 2 : **fonction linéaire avec pente** ❌

Peut être utilisée comme classificateur, mais les valeurs obtenues en projetant les points sur celle ci peuvent être en dehors de notre codomaine.

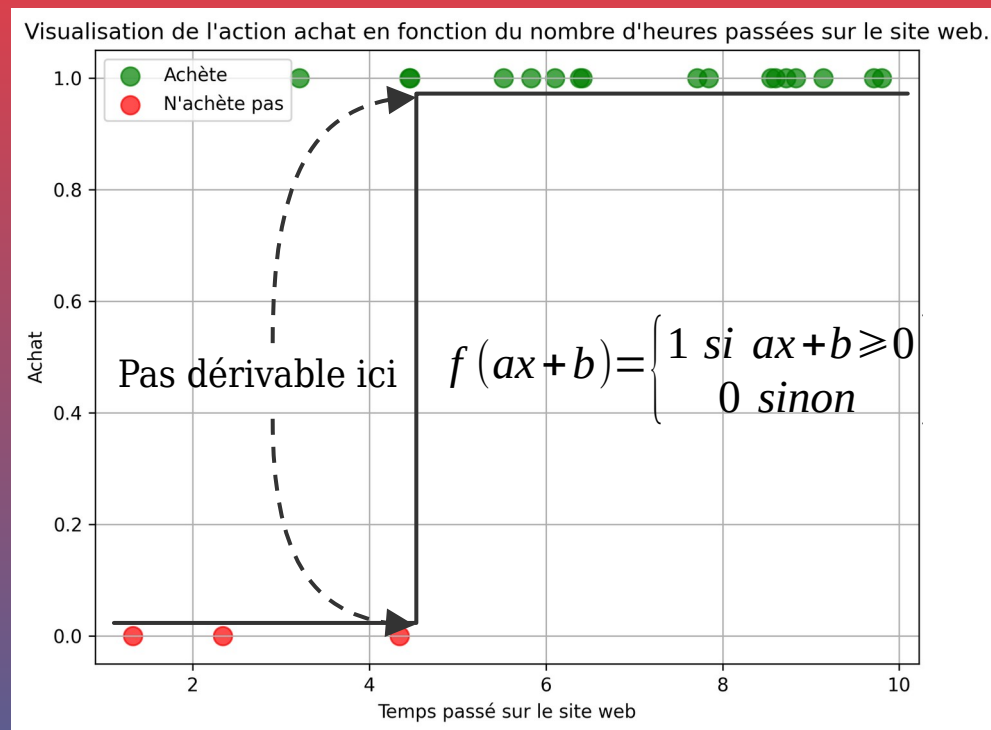
Cette valeur $\notin [0,1]$



Visualisons les données en ne considérant que le temps passé sur le site web.

Cherchons une fonction séparatrice, c'est-à-dire notre **classificateur**.

Choix 3 : **fonction échelon**



Peut être utilisée comme classificateur, mais elle présente plusieurs limites : elle n'est pas dérivable en tout point ; la transition entre 0 et 1 est trop brutale ; sa dérivée est nulle presque partout, ce qui empêche l'utilisation des méthodes d'optimisation par gradient.

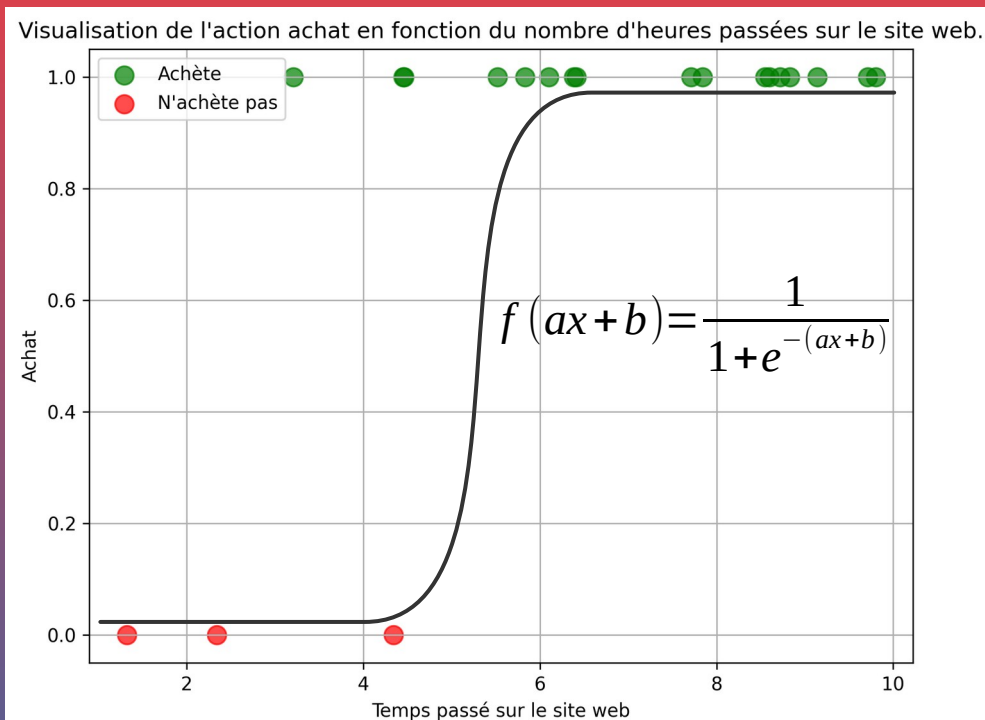
Visualisons les données en ne considérant que le temps passé sur le site web.

Cherchons une fonction séparatrice, c'est-à-dire notre **classificateur**.

Choix 4 : **fonction sigmoïde**



La fonction sigmoïde est idéale comme classificateur, car les valeurs obtenues en projetant les points sur celle-ci sont comprises entre $]0, 1[$ et elle est dérivable à chaque point.



$$P(y|\mathbf{x}; \theta) = P(y|x_1, x_2; \theta) = h_{\theta}(\mathbf{x}) = \frac{1}{1 + e^{(-\theta^T \mathbf{x})}}$$

$\theta \rightarrow$ Paramètres de notre fonction sigmoïde

$$h_{\theta}(\mathbf{x}) \in]0, 1[$$

$$\theta = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix} \rightarrow \theta^T = [\theta_1 \ \theta_2 \ \theta_3]; \quad \theta \in \mathbb{R}^3 \quad \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \xrightarrow{\text{Augmentons } \mathbf{x}} \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ 1 \end{bmatrix}$$

$$\theta^T \mathbf{x} = [\theta_1 \ \theta_2 \ \theta_3] \begin{bmatrix} x_1 \\ x_2 \\ 1 \end{bmatrix} = (\theta_1 x_1 + \theta_2 x_2 + \theta_3) \in \mathbb{R}$$

$$h_{\theta}(\mathbf{x}) = \frac{1}{1 + e^{-(\theta_1 x_1 + \theta_2 x_2 + \theta_3)}}$$

Remplaçons $\theta^T \mathbf{x}$ par son expression développée dans $h_{\theta}(x)$

(1) La probabilité que le client ait acheté

(2) La probabilité que le client n'ait pas acheté

$$P(y=1|\mathbf{x}; \theta) = h_{\theta}(\mathbf{x}) = \frac{1}{1 + e^{(-\theta^T \mathbf{x})}}$$

$$P(y=0|\mathbf{x}; \theta) = 1 - h_{\theta}(\mathbf{x}) = 1 - \frac{1}{1 + e^{(-\theta^T \mathbf{x})}}$$

Combinons (1) et (2)

$$P(y|\mathbf{x}; \theta) = (h_{\theta}(\mathbf{x}))^y (1 - h_{\theta}(\mathbf{x}))^{1-y} =$$

$$\begin{cases} h_{\theta}(\mathbf{x}); y=1 \\ 1 - h_{\theta}(\mathbf{x}); y=0 \end{cases}$$

$$\mathfrak{Z}(\theta) = \prod_{i=1}^I P(y^i|\mathbf{x}^i; \theta)$$

$I \rightarrow$ nombre de données (échantillons)

Fonction objectif à maximiser via le paramètre θ . Nous supposons que l'achat et le non-achat sont des événements indépendants.

Utilisons la fonction logarithme pour simplifier les calculs.

$$\mathcal{L}(\theta) = \log(\mathfrak{Z}(\theta)) = \log\left(\prod_{i=1}^I P(y^i|\mathbf{x}^i; \theta)\right)$$

$$\mathcal{L}(\theta) = \log \left(\prod_{i=1}^I P(y^i | \mathbf{x}^i; \theta) \right) = \sum_{i=1}^I \log (P(y^i | \mathbf{x}^i; \theta))$$

Car $\log(a * b) = \log a + \log b$

$$\mathcal{L}(\theta) = \sum_{i=1}^I \log (P(y^i | \mathbf{x}^i; \theta)) = \sum_{i=1}^I \log ((h_{\theta}(\mathbf{x}^i))^{y^i} (1 - h_{\theta}(\mathbf{x}^i))^{1-y^i})$$

$$\mathcal{L}(\theta) = \sum_{i=1}^I \log (P(y^i | \mathbf{x}^i; \theta)) = \sum_{i=1}^I \log (h_{\theta}(\mathbf{x}^i)^{y^i}) + \log ((1 - h_{\theta}(\mathbf{x}^i))^{1-y^i})$$

$$\mathcal{L}(\theta) = \sum_{i=1}^I y^i \log (h_{\theta}(\mathbf{x}^i)) + (1 - y^i) \log (1 - h_{\theta}(\mathbf{x}^i))$$

Car $\log(a^b) = b \log a$

$$\mathcal{L}(\theta) = \sum_{i=1}^I y^i \log(h_{\theta}(\mathbf{x}^i)) + (1 - y^i) \log(1 - h_{\theta}(\mathbf{x}^i))$$

$$\mathcal{L}(\theta) = y \log(h_{\theta}(\mathbf{x})) + (1 - y) \log(1 - h_{\theta}(\mathbf{x}))$$

Nous recherchons les paramètres θ qui maximisent $\mathcal{L} \rightarrow \arg \max_{\theta} \mathcal{L}(\theta)$



Maximiser $\mathcal{L}(\theta)$ est équivalent à minimiser $-\mathcal{L}(\theta) \rightarrow \arg \min_{\theta} -\mathcal{L}(\theta)$

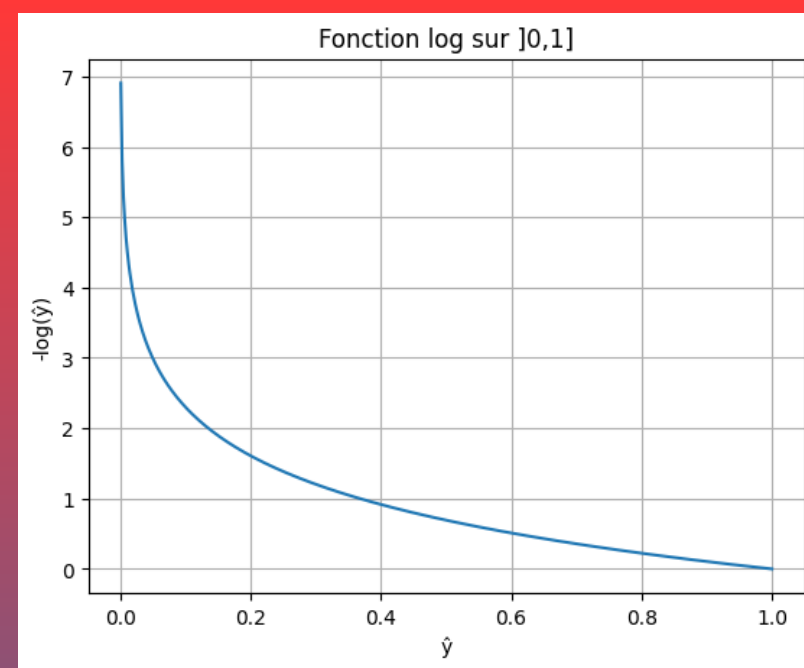
Donc avec $I > 1$ notre nouvelle fonction objective sera :

$$\mathcal{L}(\theta) = - \sum_{i=1}^I y^i \log(h_{\theta}(\mathbf{x}^i)) + (1 - y^i) \log(1 - h_{\theta}(\mathbf{x}^i))$$

Posons $\hat{y} = h_{\theta}(\mathbf{x}^i)$

$$\mathcal{L}(\theta) = - \sum_{i=1}^I y^i \log(\hat{y}) + (1 - y^i) \log(1 - \hat{y})$$

Différents scénarios de prédiction



Si la valeur réelle $y=1$ et que \hat{y} est proche de 1, le modèle est moins pénalisé car $-\log(\hat{y})$ aura une petite valeur.
 $\hat{y} \rightarrow 1$

Si la valeur réelle $y=0$ et que \hat{y} est proche de 0, le modèle est plus pénalisé car $-\log(\hat{y})$ aura une grande valeur.
 $\hat{y} \rightarrow 0$

Si la valeur réelle $y=0$ et que \hat{y} est proche de 1, le modèle est plus pénalisé car $-\log(1 - \hat{y})$ aura une grande valeur.
 $(1 - \hat{y}) \rightarrow 0$

Si la valeur réelle $y=0$ et que \hat{y} est proche de 0, le modèle est moins pénalisé car $-\log(1 - \hat{y})$ aura une petite valeur.
 $(1 - \hat{y}) \rightarrow 1$

Minimiser la fonction de perte consiste à trouver un paramètre θ tel que le gradient $\nabla_{\theta} \mathcal{L}_{\theta}(y, \hat{y})$ de la fonction objective (fonction de perte) par rapport à θ soit nul.

$$\mathcal{L}(\theta) = \mathcal{L}_{\theta}(y, \hat{y}) = - \sum_{i=1}^I y^i \log(\hat{y}) + (1 - y^i) \log(1 - \hat{y})$$

Donc on aura: $\nabla_{\theta} \mathcal{L}_{\theta}(y, \hat{y}) = 0$ et $\partial_{\theta_n} \mathcal{L}(\theta_n) = 0$

Appliquons la règle de la chaîne (the chain rule) $\rightarrow \frac{\partial \mathcal{L}}{\partial \theta_n} = \frac{\partial \mathcal{L}}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial z} \frac{\partial z}{\partial \theta_n}$

$$\text{Avec } \hat{y} = \frac{1}{1 + e^{-z}}; z = \theta^T \mathbf{x}; \theta^T = [\theta_1 \theta_2 \theta_3]$$